

Building Something That Failed

During my second year on the school robotics team, I volunteered to lead the design of a small automated sorting machine. The goal appeared simple. The machine needed to identify colored objects on a conveyor belt and separate them into different containers. Our team believed the project would perform well in a regional engineering competition.

At the start, the design phase moved smoothly. I studied previous robotics projects and assembled a small group of teammates interested in programming and hardware. We divided responsibilities. One student built the frame. Another wrote the initial control code. I focused on sensor calibration and system coordination.

The first tests looked promising. The machine identified colors correctly and directed objects to the correct containers. Confidence grew within the team. We believed the system would operate reliably during the competition.

The competition day revealed our mistake. The lighting inside the exhibition hall differed from the lighting in our classroom. The color sensor struggled to distinguish between objects under the new conditions. The machine misclassified several items during the first demonstration round.

At first I searched for a quick technical fix. After a few minutes I recognized a deeper issue. Our design depended too heavily on a single sensor. We had not prepared alternative detection methods or tested the system under different environmental conditions.

After the event our team met to review the project. Instead of focusing on the loss, we analyzed the design process. We introduced a new testing routine that required experiments under multiple

lighting conditions. The following season our team developed a new system using additional sensors and image processing.

The original machine never reached the performance we expected. Yet the failure produced a more valuable lesson. Careful testing and flexible design shape reliable engineering solutions.